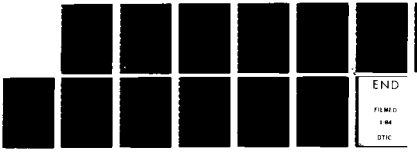
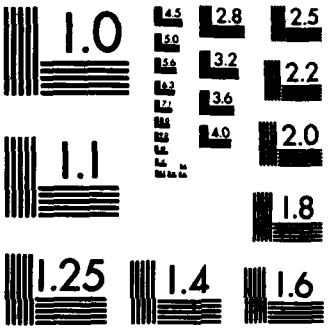
AD-A135 725 A STRUCTURED DESIGN METHODOLOGY FOR VLSI SYSTEMS(U) 1/1
STANFORD UNIV CA STANFORD ELECTRONICS LABS R W DUTTON
NOV 83 ARO-17200.8-EL DAAG29-80-K-0046

UNCLASSIFIED F/G 9/5 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

A STRUCTURED DESIGN METHODOLOGY FOR VLSI SYSTEMS

Final Report

by

Robert W. Dutton and Willem VanCleemput
Stanford Electronics Laboratory
Stanford University
Stanford, CA 94305

August 1, 1980 - July 31, 1983

Prepared for

U. S. Army Research Office
Research Triangle Park, NC

Contract No. DAAG29-80-K-0046

Approved for public release; distribution unlimited

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. A135724	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A STRUCTURED DESIGN METHODOLOGY FOR VLSI SYSTEMS		5. TYPE OF REPORT & PERIOD COVERED Final Report August 1980-July 31, 1983
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Robert W. Dutton		8. CONTRACT OR GRANT NUMBER(s) DAAG29-80-K-0046
9. PERFORMING ORGANIZATION NAME AND ADDRESS Stanford Electronics Laboratory Stanford University Stanford, CA 94305		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS U. S. Army Research Office Post Office Box 12211 Research Triangle Park, NC 27709		12. REPORT DATE November 1983
		13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

The view, opinions, and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Hierarchical design, integrated circuits, functional simulation, layout, verification, design rule checking, routing, compaction.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

New algorithms and tools for hierarchical design and verification of IC chips have been developed. The multi-level simulation capabilities of SABLE have been extended using the ADA language. Automatic synthesis of stick diagrams from net lists and subsequent pitch-constrained compaction of layout provide new capabilities for cell generation. New parallel algorithms for routing and design rule checking have been developed which are orders-of-magnitude faster than conventional approaches. Analytic models to accurately estimate and bound waveforms in MOS circuits have been developed and provide new insight for performance enhancement.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

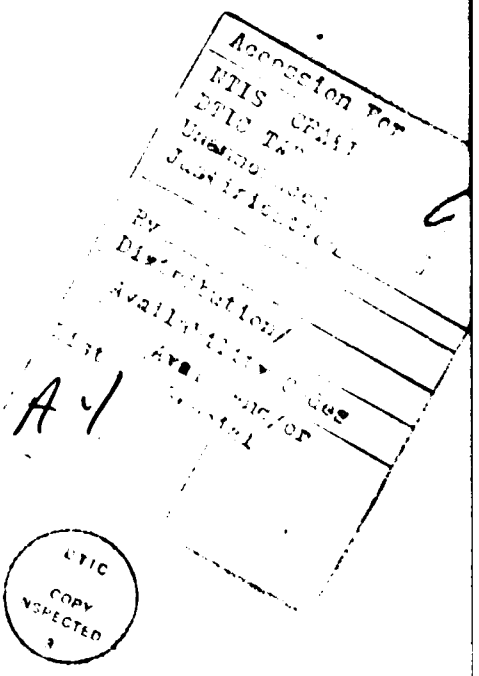
UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

A STRUCTURED DESIGN METHODOLOGY FOR VLSI SYSTEMS

Abstract

New algorithms and tools for hierarchical design and verification of IC chips have been developed. The multi-level simulation capabilities of SABLE have been extended using the ADA language. Automatic synthesis of stick diagrams from net lists and subsequent pitch-constrained compaction of layout provide new capabilities for cell generation. New parallel algorithms for routing and design rule checking have been developed which are orders-of-magnitude faster than conventional approaches. Analytic models to accurately estimate and bound waveforms in MOS circuits have been developed and provide new insight for performance enhancement.



1. Introduction

The design and implementation of very large scale integrated circuits requires new approaches and tools. This research project has developed a high level language oriented environment for IC specification/verification as well as novel tools and methodology for IC layout. The effort has supported the completion of five PhD theses, three more are expected within six months and two new dissertation topics have been defined. The scope of the contract covered both high level specification and simulation as well as a variety of implementation problems including layout and verification. In each of these areas, specific goals were proposed. The most important of these are now summarized along with actual accomplishments.

2. Hardware Design Specification

1. Unify behavioral description with emphasis towards design verification
2. Couple multilevel behavior description with multilevel simulation capabilities.
3. Integrate behavioral and structural description into a coherent design methodology.

Other proposed tasks related to specific language enhancements for SDL and DDL and design system interfaces and interactive graphics were pursued to a lesser extent as discussed below.

2.1 Accomplishments

In the areas of activity outlined above, several major accomplishments have been achieved. First, the SABLE simulator effort using the ADLIB language has culminated in a major contribution to high level simulation [1]. The program has been distributed to more than 18 academic and industrial locations and was cited specifically in the VHSIC phase III specification as a good example of a high level design language. A spin-off from this original work has addressed formal verification issues [2]. A second major accomplishment is the use of ADLIB-SABLE for fault simulation [3] and more recent work in verification using an ADA-based extension of SABLE [4]. Finally, the demonstration of an isomorphically constrained design methodology [5] has shown that consistency between schematics and cell-based layout can be maintained using proper

interfaces of editors and a novel means to update the data-base. In the following paragraphs the key features of each of these accomplishments is reviewed.

A major focus and thrust of this project is to exploit hierarchy and develop the necessary tools to unify the various levels of abstraction used in a complex digital IC design. The results of Dwight Hill in his PhD thesis [1] demonstrated that a PASCAL-based language, ADLIB, could be used effectively to express functional behavior description and evaluated with the SABLE simulator. SABLE is a multi-level general-purpose simulator which views the world as a collection of components interconnected through well defined nets; the nets can assume any valid PASCAL data type. This software has been distributed widely among DoD vendors and the concepts are indeed well received. Intel has adopted a very similar approach using an ADA-based language. From the kernel idea of ADLIB-SABLE, two follow-on efforts have had a major impact and will now be discussed.

Functional verification is a difficult problem and substantial progress has now been achieved using extensions of the ADLIB work. Given two descriptions of a digital design, their comparison to prove design correctness is indeed difficult. Cory has demonstrated a symbolic simulation [2] based on ADLIB and SDL descriptions of multiple representations. The verification efforts have revealed that the use of non-selective trace for combinational components provides increased efficiency since extensive case-splitting can be avoided. In addition, a new concept of case merging can reduce both storage and computation. For example if components A and B split into M and N cases, the conventional simulations using splitting require MN3 cases whereas by using merging the number is reduced to 3(M+N). Moreover, the merging technique allows all cases to be simulated automatically---a potential advantage if custom hardware is used. The only drawback is the trade off to be made in complexity of expressions that must be evaluated using case merging in contrast to the exhaustive simplicity of case splitting.

Further studies of fault simulation have followed based on the constructs of SABLE. Concurrent fault simulation is another technique which has been compared with the previously reported deductive fault simulation. Comparisons have revealed that

concurrent modeling approach is more general. Specifically, it can handle multi-valued cases as is typical for MOS technology. Further studies are in progress to characterize both approaches based on SABLE implementations. The choice of verification method, either by case splitting or merging has been evaluated and the results suggest that in many cases the increased complexity of expressions resulting from case merging causes a proportional increase in analysis time which offsets any advantages. While for some cases merging is of value, we cannot draw any definitive conclusions. In the area of consistency checking, the assumptions and constraints of each design approach are critical. A careful theoretical look at previous work as well as new results developed in this contract have yielded a criterion for deciding if a design refinement is correct. The results apply to deterministic or nondeterministic hardware descriptions--with or without don't cares.

The second set of activities related to ADLIB-SABLE regards functional and fault simulation [3]. Fault simulation yields results that are useful in diagnosing a manufactured system design or assisting in a better design from fault tolerance point of view. The descriptive power of ADLIB provides unique leverage concerning logic descriptions---it is simple and does not require recompilation for each new circuit. Instead the SDL description is modified and compiled prior to simulation. The result is a streamlined specification and efficient simulation tool. The area of deductive fault simulation has been pursued using ADLIB-SABLE. The power of ADLIB for simplified description is attractive and the power of deductive fault simulation should yield improved results over parallel techniques due to better fault coverage. Presently the models include stuck-at 0, 1, and unknown.

The fault simulation results using ADLIB-SABLE have yielded a detailed comparison of deductive simulation with conventional simulators such as SALOGS [6]. Results show that the increase in CPU time with the number of gates fault simulated, is linear in contrast to a quadratic dependence for SALOGS. The deductive and concurrent approaches are shown to be equivalent although there may be performance advantages for concurrent simulation given a multi- processor environment.

When doing IC design, it is convenient to work simultaneously with different representations of a circuit. A schematic is needed for human functional comprehension, whereas an IC artwork representation is needed to implement the design in silicon. Isomorphic representations have the same circuit topology, and it is relatively easy to check consistency between them. An isomorphic design system was implemented along with its application in the design of a 2400 transistor circuit [5]. The key components of the system include a module update manager (MUM) and a graphics editor implemented on an HP9845C used for both schematics and layout. The MUM maintains consistency by passing messages between modules which indicate to the user what needs to be done to make the design consistent. Messages are passed between different levels in the hierarchy to maintain hierarchical consistency as well as between isomorphic representations of the same module.

The notable success of the system was maintaining consistency of a hierarchy of more than 40 modules, some instantiated as many as 132 times, with nesting as much as 7 deep. Hierarchical consistency was maintained by: (1) sending messages to all owners of a module whenever that module's interface changed, (2) hierarchical port extraction, and (3) incremental connectivity analysis to insure that cells fit together correctly. Maintaining consistency between isomorphic representations was demonstrated to work for higher level representations such as logic diagrams. However, it was found that the simple design system was insufficient for maintaining isomorphism between logic diagrams and IC artwork, since no provision was made for gate geometry depending on context as is common in IC network. The results suggest that different geometrical versions of gates are necessary, perhaps by using parameterized cells. This result corraborates the suggested layout methodological constraint to be discussed in the next section.

3. Hierarchical Layout

1. Study of placement and routing algorithms for IC subblocks with arbitrary shaped size.

2. Development of algorithms to optimize the aspect ratio of rectangular macrocells.

3. Develop tools to facilitate reconstruction of functional level information from IC topological inputs.

Although the S-1 processor design was originally proposed as a study vehicle, applications based on cells and system designs typical of the Mead-Conway style design course [7] were found to be more suitable. The gate array and hybrid efforts originally proposed were not pursued further owing to intense industrial developments in these areas.

3.1 Accomplishments

In the areas of activities outlined above, several major accomplishments have been realized. The thesis work of Preas [8] and Slutz [9] was the starting point for hierarchical layout work discussed here. Preas attached primarily the routing problems and used graph theoretical methods to guarantee 100% routing completion as demonstrated with the SANDIA program called SICLOPS. Slutz emphasized placement and using a "wall-model" from architecture developed a vocabulary and tools for synthesis of layout. The background of these two efforts lead to three alternative approaches to the problem. First, Beetem [5] demonstrated an editor-based manual layout system with special emphasis on schematic consistency and ability to edit layouts, even those generated by automated tools. Second, Scheffer developed an approach based on strict hierarchy [10] to insure certain layout constraints, especially design rule checking (DRC). Moreover, he demonstrated a novel capability to attach programs to graphics to create parameterizable cells in a hierarchical way. Finally Wolf [11] addressed the problem of layout synthesis directly from the specification of a component wire list. The DUMBO [11] program automatically generates stick diagrams and the LAVA program applies "supercompaction" algorithms [12] to achieve pitch-constrained optimal layout. Further details of each of these activities is now summarized.

John Beetem [5] demonstrated that using multiple editors--one from schematics and another for layout---one could ensure isomorphism using a novel incremental update manager scheme as discussed above. The approach has great appeal even in the context of automated tools such as MP2D [13] since it is possible to subsequently edit the designs with guaranteed correctness.

The isomorphic design methodology was tested for a standard cell- based design. Using logic schematic entry as the master representation, a cell-based layout was generated by hand. The MUM system guaranteed that all interconnections of the cell layout satisfied the original logic schematics. A key point to emphasize here is that it is reasonable to expect that even by using automatic place and route programs such as MP2D [13], the MUM concept would allow for manual intervention and editing while still guaranteeing that the logic schematic intent was not violated.

Parameterization of designs is a second major area of ongoing activity under this contract. It has been demonstrated that layouts can be parameterized using the combination of programs and graphics [10]. Similar isomorphic design efforts, the need to exploit multiple representations requires care in implementation. However, the use of program-generated layout can substantially ease the layout burden-- especially when designs must be implemented in a new technology or scaled. This also gives leverage in creating reusable layout designs since both function and layout implementation have been captured in a single high level form.

Using a combined schematic-like drawing tool as the primitive tool, two key innovations have been added. First, a language attachment has been demonstrated whereby a Pascal-like program can create schematics based on recursive language constructs. The functional intent of the blocks including special cases -- for example carry chain terminations in bit slice architectures--are automatically handled. A specific approach to layout implementation facilitates direct translation of schematic to layout and avoids full chip DRC. These advantages are achieved by requiring non-overlapping cells, new cell instances for each modification and a well-specified boundary "bumper" for DRC.

The automatic generation of stick diagrams and compacted layout for IC building blocks provides unique potential, especially for components not tightly constrained in space or performance. For example, many peripheral functions such as registers and drivers are easily specified and are not tightly constrained. A prototype program [11] now automatically generates stick layouts based on net list specifications. By means of a number of test examples, several aspects of the compiler and its interactions with the

layout compactor are more precisely understood. Results show that a fully automated approach gives 120% area penalty compared to the hand layout while by giving hints at the expanded and router levels the area penalty is reduced to 60% and 15% respectively. For many applications the fully automated approach may be acceptable. Conversely, he interactive approach and evaluation of the role of hints in the process provides useful insight into the design process.

4. Algorithms for Layout Synthesis and Verification

In addition to the high level language work in layout discussed above, new efforts have been directed at the "back-end" problems of routing, rule checking and electrical verification. While these efforts were not originally cited in the proposal, significant progress has been made and the results are now summarized. The two areas to be discussed are:

1. Parallel algorithms and hardware for layout
2. Determination of timing bounds for MOS tree networks

4.1 Accomplishments

The hierarchical specification and layout of IC dhips is often referred to as a "top-down" approach. The actual implementation of IC's at the layout level finally produces patterns of gates and interconnections-- both passive and active--which lead to mask generation and finally chip fabrication. In the final stages of IC implementation there is an explosion of information since abstractions such as wires and gates become physically instantiated. The efforts now discussed provide major leverage in solving the problems of handling this layout related data in an efficient and systematic way. The PhD thesis work of Tom Blank has developed parallel algorithms [14, 15] and a novel bit map hardware approach [15] to solve problems such as routing and design rule checking (DRC). The PhD thesis of Mark Horowitz [16] addresses the performance aspects of chip design in terms of determining timing bounds for arbitrary MOS tree structures--a dominant form of layout-oriented structures for VLSI.

The opportunity to exploit parallelism in algorithms and computer architectures has

been an elusive goal--except for specific applications. IC layout offers an exceptional opportunity to apply parallelism since thousands of gate structures are to be placed and routed on a non-overlapping two-dimensional surface of a chip. The thesis work of Blank [15] has uniquely demonstrated the potential advantages of parallelism for IC layout tasks such as wire routing and design rule checking. In the case of routing, the well-known Lee Algorithm [17] which propagates wave fronts to find the shortest path between wiring pins is naturally applied in a parallel approach. Each wave front is created by the algorithm with a single instruction so that after repeated application of this instruction the routing path can be traced. Design rule checking is another major layout bottleneck. Commonly used algorithms [18] involve expansion of features in the layout and subsequent checking of layer spacings to determine if violations have occurred. Blank has demonstrated that these algorithms can be efficiently implemented in a parallel bit map approach so any rule check can be performed on all features (simultaneously) by a small programming sequence. Both the routing and design rule checking algorithms have been simulated and show two to three orders-of-magnitude speed improvement over serial approaches presently used. In order to in fact realize these advantages of parallelism, hardware must be available to execute the algorithms. Blank has designed such a hardware system [15] and a prototype system is now being constructed at Stanford.

The electrical verification of performance of IC chips involves two key steps:

1. Extraction of parameters such as resistance, capacitance and transistor data and

2. Simulation of waveforms with simulators such as SPICE [19] or SALOGS [6].

Current DRC programs [18] provide adequate facilities for parameter extraction. While simulation provides essential performance information, it is costly for large circuits and does not provide qualitative output to identify critical design problems. Horowitz has developed new analytical approaches [16] to bound transient voltage waveforms as would be produced by complete SPICE simulation. These analytic solutions and bounds not only match simulation but also provide quantitative means to deduce critical paths in the design since the time constants and bounds of each waveform are directly related to

extracted component values. The models include accurate evaluation of transistor nonlinearities--both gates and pass-transistor networks. The effects of ramping rate at inputs is also modeled. This latter feature, in contrast to models used in logic simulators, is essential for MOS circuits where the current drive capabilities with capacitive load are limited.

5. Conclusions

This research project has addressed several aspects of tools and high level language approaches for hierarchical design of VLSI chips. Behavior description, simulation and verification efforts have resulted in the ADLIB language and SABLE simulator. This work is widely recognized as a fundamental contribution to the simulation area and has directly impacted the DoD development of VHDL. Follow-on work at Stanford has demonstrated new capabilities based on the ADA language and efforts in this direction will continue. In the area of specification a novel data management scheme (MUM) was demonstrated based on enforced isomorphism between levels of abstraction--for example schematics and layout. This approach is especially attractive for cell-based layout such as MP2D; allowing manual editing without producing functional errors.

In the area of layout several major contributions have been discussed. A new layout synthesis approach has been demonstrated using automatic generation of stick diagrams directly from logic and net lists. The program (DUMBO) produces symbolic layout which in turn is compacted to produce actual mask layout. In addition, new compaction algorithms have been developed to minimize layout pitch by resolving x-y interlocks. Another contribution to the layout area is the advances in coupling programs and graphics. The ability to synthesize blocks and modify them by programming constructs has been demonstrated with direct graphical display. This layout technique allows custom configurations of cells to be programmed and modified with design rules and other technology-based features becoming transparent to the user.

New algorithms for routing and design rule checking have been developed using the parallelism of a custom bit map processor architecture. These algorithms show orders-of-magnitude speed advantage over serial approaches on single CPU's. A hardware

architecture to realize these advantages has been designed and is now being built. Finally, a new set of analytic models have been developed for bound timing waveforms in MOS circuits. These algorithms closely match SPICE waveforms, require a minimum computational effort and provide insight concerning performance limits which is not obtained from simulation data alone.

References

[1] D. D. Hill, "Language and Environment for Multilevel Simulation", Ph.D. Dissertation, Stanford University, March 1980

[2] W. E. Cory, "Verification of Hardware Design Correctness: Symbolic Execution Techniques and Criteria for Consistency," Ph.D. Dissertation, Stanford University, June 1983.

[3] S.Ghosh and W. VanCleemput, "A Generalized Timing Verifier," Submitted to the Transactions on Computer Aided Design.

[4] S. Ghosh and W. VanCleemput, "Fault Simulation in ADLIB-SABLE," Technical Report No. 83-242, Computer Systems Laboratory, Stanford University, March 1983.

[5] J. F. Beetem, "Structured Design of Electronic Systems Using Isomorphic Representations," PhD thesis, Stanford University, December 1981.

[6] J. M. Acken and J. D.Stauffer, "User's Manual Sandia Logic Simulator (SALOGS) Network Program, Technical Report SAND81-0080.

[7] C. Mead and L. Conway, "Introduction to VLSI Systems, Reading, MA: Addision-Wesley, 1980.

[8] B. T. Preas, "Placement and Routing Algorithms for Hierarchical Integrated Circuit Layout," Ph.D. Dissertation, Stanford University, August 1979.

[9] E. A. Slutz, "Shape Determination and Placement Algorithms for Hierarchical Integrated Circuit Layout," PhD Dissertation, Stanford University, August 1980.

[10] L. K. Scheffer, "A Methodology for Improved Verification of VSLI Designs Without Loss of Area," Presented at Second Caltech Conference on Very Large Scale Integration, January 1981.

[11] W. Wolf, J. Newkirk, R. Mathews, "DUMBO, A Schematic-to- Layout Compiler," Presented at Second Caltech Conference on Very Large Scale Integration, January 1981.

[12] W. Wolf, R. Mathews, J. Newkirk, and R. W. Dutton, "Two- Dimensional Compaction Strategies," ICCAD Proceedings, Santa Clara, CA, Sept. 1983.

[13] A. Feller, "Automatic Layout of Low-Cost Quick-Turnaround Random- Logic Custom LSI Devices," Proceedings 13th Design Automation Conference, San Francisco, California, June 1976, pp. 79-85.

[14] W. T. Blank, M.Stefik, and W. VanCleemput, "A Parallel Bit Map Processor Architecture for DA Algorithms," Proceedings 18th Design Automation Conference, Nashville, June 1981.

[15] W. T. Blank, "A Bit Map Architecture and Algorithms for Design Automation," Ph.D. Dissertation, Stanford University, Sept. 1982

[16] M. Horowitz, "Timing Models for MOS Circuits," PhD Dissertation, Stanford University, To be published December 1983.

[17] C. Y. Lee, An Algorithm for Path Connections and Its Applications, IRE Trans. Electron Compouters EC-10:346-365, Sept. 1961.

[18] D. Alexander, "A VLSI Design Rule Checker," In International Conference on Computer Aided Design and Manufacture of Electronic Components, Circuits and Systems, pp. 57-60, IEE, July 1979.

[19] L. W. Nagel, "SPICE2: A Computer Program to Simulate Semiconductor Circuits," Electron Re. Lab., University of California, Berkeley, CA, Memo No. ERL-M520, May 1975.

Scientific Personnel Supported by this Contract

Robert W. Dutton, Principal Investigator
Professor, Electrical Engineering Department
Stanford University

Willem VanCleemput, Co-principal Investigator

John Beetem
(Received PhD degree, Autumn Quarter 1981)

William Thomas Blank
(Received PhD degree, Autumn Quarter 1982)

Warren Cory
(Received PhD degree, Winter Quarter 1982)

Sumit Ghosh
(Received MS degree, Autumn Quarter 1981,
Expected to receive PhD degree, Winter 1983)

Mark Horowitz
(Expected to receive PhD degree, Winter 1983)

Lou Scheffer
(Passed PhD Orals, Spring 1981,
Expected to receive PhD degree, Winter 1983)

Donald E. Ward
(Received PhD degree, Spring 1981)

Wayne Wolf
(MS received Spring Quarter 1981)
(Expected to receive PhD degree, Winter 1983)

Youm Huh
Research Assistant
Stanford University

Hussein Al-Hussein
Research Assistant
Stanford University

Technical Reports and Publications Sponsored under this Contract

(1) W. E. Cory, "Symbolic Simulation for Functional Verification with ADLIB and SDL," Proceedings 18th Design Automation Conference, Nashville, June 1981, pp. 82-89.

(2) R. W. Dutton, "Stanford Overview in VLSI Research," IEEE Trans. on Circuits and Systems, Vol. CAS-28, No. 7, July 1981, pp. 654-665.

(3) R. W. Dutton, "Position Statement--Tools for Design Automation from a University Point of View," Proc. 18th Design Automatin Conference, Nashville, June 1981, p. 333.

(4) W. T. Blank, M. Stefik, and W. VanCleemput, "A Parallel Bit Map Processor Architecture for DA Algorithms," Proceedings 18th Design Automation Conference, Nashville, June 1981, pp. 837-844.

(5) D. E. Ward, "Charge-Based Modeling of Capacitance in MOS Transistors," PhD Dissertation, Stanford University, June 1981.

(6) J. F. Beetem, "Structured Design of Electronic Systems Using Isomorphic Multiple Representations, PhD Dissertation, Stanford University, December 1981.

(7) H. K. Al-Hussein, and R. W. Dutton "Path Delay Computation for Integrated Systems, ICCC'82, New York, September 1982.

(8) W. T. Blank, "A Bit Map Architecture and Algorithms for Design Automation, PhD Dissertation, Stanford University, September 1982.

(9) W. Wolf, J. Newkirk, R. Mathews, "DUMBO, A Schematic-to-Layout Compiler," Presented at Caltech Conference on VLSI, March 1983.

(10) Sumit Ghosh and Willem VanCleemput, "Fault Simulation in ADLIB-SABLE," Technical Report No. 83-242, Center For Integrated Systems, Computer Systems Laboratory, Stanford University, March 1983.

(11) W. E. Cory, "Verification of Hardware Design Correctness: Symbolic Execution Techniques and Criteria for Consistency," PhD Dissertation, Stanford University, June 1983.

(12) J. Rubinstein, P. Penfield, M. Horowitz, "Signal Delay in RC Tree Networks," IEEE Transactions on Computer Aided Design, June 1983.

(13) W. Wolf, R. Mathews, J. Newkirk, and R. W. Dutton, "Two-Dimensional Compaction Strategies," To be presented ICCAD, Santa Clara, September 1983.

(14) S. Ghosh and W. VanCleemput, "Functional Fault Simulation Using ADLIB-SABLE," To be presented ICCAD, Santa Clara, September 1983.

(15) M. Horowitz, "Timing Models for MOS Pass Nets," ISCAS 1983.

(16) M. Horowitz, R. W. Dutton, "Resistance Extraction from Mask Layout Data," IEEE Transactions on Computer Aided Design, To be published August 1983.

(17) S. Ghosh and W. VanCleemput, "A Generalised Timing Verifier," Submitted to the Transactions on Computer Aided Design.

END

FILMED

1-84

DTIC